AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions of claims in the application:

Listing of Claims:

1. (Currently amended) A user interface automation system comprising:

a processor that executes the following computer executable components stored in memory:

an input component that receives a request; ~~and~~ [[,]]

a navigation component that receives the request from the input component, and facilitates ~~simulated~~ simulating a user interface associated with an automation component; ~~based, at least in part, upon information stored in~~

a map information store comprising program flow information wherein one or more section names divide the map information store into specific page data such that each of the section name references a specific page of the automation component;

a command information store disparate from the map information store that comprises the section names associated with the specific page of the automation component and one or more actions to execute for the specific page;

~~and~~ wherein the navigation component facilitates simulating the user interface based at least upon information stored in [[a]] the command information store[[,]] and the map information store, wherein the map information store, the command information store and associated executables are stored separately, the navigation component further modifies the user interface ~~automation~~ without recompiling executables ~~by modifying~~ upon modification of one or more of the map information store or the command information store, the navigation component further employs information stored in a global information store when a global variable is encountered in the command information store and facilitates a global variable replacement from a single location and sharing of a common program flow among a plurality of users.

2. (Original) The system of claim 1, wherein the automation component is a wizard.

3. (Original) The system of claim 1, wherein the map information store comprises a text-based file.

4. (Previously presented) The system of claim 1, wherein the command information store comprises a text-based file.

5. (Cancelled)

6. (Cancelled)

7. (Currently Amended) The system of claim 1, wherein at least one of the map information store ~~and~~ or the configuration information store comprise at least one alias name.

8. (Original) The system of claim 1, wherein the navigation component further stores error information in a log information store.

9. (Original) The system of claim 1, wherein the navigation component further stores information associated with the request in a log information store.

10. (Original) The system of claim 9, wherein the navigation component iterates through information stored in the command information store, performs the indicated operation and stores information associated with the indicated operation in the log information store.

11. (Original) The system of claim 9, wherein the navigation component stores error information in the log information store.

12. (Original) The system of claim 1, wherein the input component performs input validation upon the request and provides error information if the request is invalid.

13. (Original) The system of claim 12, wherein a graphical message is displayed to a user of the system, the graphical message being based, at least in part, upon the error information from the input component.

14. (Original) The system of claim 1, wherein the input component receives a command line invocation.

15. (Currently Amended) The system of claim 1, the map information store comprises ~~comprising a section name and~~ a page identifier.

16. (Currently Amended) The system of claim 15, the page identifier comprising a label for a control, the page identifier further uniquely identifying ~~a particular~~ the specific page.

17. (Original) The system of claim 15, the page identifier comprising a control type.

18. (Currently Amended) The system of claim 17, wherein the control type is at least one of button, combo, list, scroll, static, radio ~~and~~ or check.

19. (Currently Amended) The system of claim 1, wherein information stored in the command information store can be modified by at least one of a front-end user interface application, scripting, a batch file ~~and~~ or a text editor.

20. (Currently Amended) The system of claim 1, the command information store comprising a section name, the section name corresponding to information stored in the map information store[[,]] ~~the command information store further comprising an action~~.

21. (Currently Amended) The system of claim 1, the command information store storing information associated with at least one of a function key ~~and~~ or a control key simulation.

22. (Currently Amended) A method of automating user interface comprising:

employing a processor to execute computer executable instructions stored in memory to perform the following acts:

receiving a request invoking a user interface automation system;

storing information related to specific pages of the automation system and corresponding acts to execute for the pages in a disparate map information store and a command information store respectively;

receiving mapping information from [[a]] the map information store comprising one or more section names that reference specific pages of the automation system;

receiving command information from [[a]] the command information store comprising specific section names corresponding to information stored in the map information store and information associated with commands to be executed for respective pages of the automation system;

retrieving global information from a global information store;

~~performing simulated~~ simulating a user interface based, at least in part, upon information stored in the map information store and the command information store;

employing information stored in the global information store when a global variable is encountered in the command information store; and

modifying the user interface automation utilizing existing compiled executables upon modification of one or more of the map information store or the command information store, by storing data, commands associated with generating the user interface and the executables separately ~~and maintaining compiled executables~~.

23. (Original) The method of claim 22, further comprising:

storing information in a log information store, if an error is detected performing the simulated user interface.

24. (Original) A computer readable medium having stored thereon computer executable instructions for carrying out the method of claim 22.

25. (Currently Amended) A method of automating user interface comprising:

employing a processor to execute computer executable instructions stored in memory to perform the following acts:

receiving a request for executing a user interface automation component;

retrieving mapping information from a map file comprising at least a section name and a page identifier for pages generated by the automation component;

retrieving commands to be executed for the pages ~~information~~ from a command file;

retrieving global information from a global file;

obtaining a section name from the command file;

retrieving page identification information from the map file associated with the section name;

retrieving section data for section associated with the section name from the command file;

~~performing an action~~ executing the commands associated with the retrieved section data;

employing information stored in the global file when a global variable is encountered in the command file;

sharing a common program flow among a plurality of users;

modifying the user interface automation ~~by modifying~~ when one or more of the command file or map file ~~and~~ are modified while maintaining compilation of executables; and

separately storing [[at least one of:]] the map file, the command file[[; or]] and the compiled executables.

26. (Original) The method of claim 25, further comprising:

storing information in a log file, if an error is detected performing the action.

27. (Cancelled)

28. (Currently amended) [[A]] An ~~user interface~~ automation system for a user interface comprising:

a processor that executes the following computer executable components stored in memory:

an input component that receives a request associated with generating a user interface; [[and,]]

a map information store comprising program flow information wherein one or more section names divide the map information store into specific page data such that a specific section name references a specific page of the automation component;

a command information store that comprises command information associated with one or more pages associated with the automation component and one or more actions to execute for a given page; and

a navigation component that receives the request from the input component and facilitates ~~simulated~~ generating the user interface associated with an automation component based, at least in part, upon information stored in [[a]] the map information store and information stored in [[a]] the command information store, the map information store, the command information store and executables are stored separately, the navigation component modifies the user interface automation ~~by modifying~~ when at least one of the map information store or the command information store are modified while maintaining existing compiled executables, the navigation component further employs information stored in a global information store when a global variable is encountered in the command information store and facilitates a global variable replacement from a single location and sharing of a common program flow among a plurality of users.

29. (Currently amended) [[A]] An ~~user interface~~ automation system for a user interface comprising:

means for receiving a request in connection with simulating a user interface;

means for storing mapping information comprising one or more section names that reference specific pages of the user interface;

means for storing command information disparate from the mapping information storing means, the command information storing means comprises specific section names corresponding to the one or more section names stored in the mapping means, the command information comprises at least commands to be executed for respective pages of the user interface;

means for simulating the user interface associated with an automation component based, at least in part, upon information stored in the mapping information storing means and the command information storing means ~~a map information store and information stored in a command information store~~, the means for simulating ~~receiving~~ receives the request from the means for receiving;

means for sharing a common program flow among a plurality of users based, at least in part, upon replacing a global variable in the command information store with corresponding data from a global information store;

means for modifying the user interface automation when one or more of the mapping information storing means or the command information storing means is modified, while maintaining compiled executables as mapping information, command information and the compiled executables are stored separately.[[;]] ~~and~~

~~means for separately storing map information data, command information data or the compiled executables.~~

30-32. (Canceled)

33. (Previously presented) The method of claim 22, wherein data and commands associated with program flow are stored in a text file.